

Claims

- [c1] A method of migrating a Lotus Notes Domino document to a non-Domino web server having a client end and a server end, comprising:
implementing a series of fetches for a Lotus Notes Domino document having a plurality section levels;
creating a single document having all section data; and
expanding and collapsing said section data at said client end.
- [c2] The method of claim 1 including performing 1+N fetches, one for each level of sections.
- [c3] The method of claim 1 wherein said single document includes a plurality of pages equal to said plurality of section levels.
- [c4] A method of migrating a section-rich Domino document to a non-domino web server having a client end and a server end, comprising:
performing iterative recursive intelligent fetch process steps;
performing a recursive differential process step; and
expanding and collapsing said section data at said client

end.

- [c5] The method of claim 4 wherein said iterative recursive intelligent fetch process includes:
 - parsing said Domino document;
 - identifying unexpanded sections; and
 - constructing a URL to expand said unexpanded sections.
- [c6] The method of claim 4 wherein said recursive differential process step includes:
 - inputting expanded section data at depth N expansion;
 - inputting expanded section data at depth N-1 expansion;
 - comparing pre- and post-expanded section documents through differences in html;
 - creating delta html files based on said comparison;
 - adding javascript conditionals around said delta html files; and
 - merging said delta html files into said depth N-1 expansion.
- [c7] The method of claim 4 including instructing Domino to expand each section by an "expandsection" html query.
- [c8] The method of claim 7 wherein said expandsection html query includes said html query for a plurality of section expansions.

- [c9] The method of claim 8 including said html query of a form: "/asdasd&ExpandSection=1,2,3,1.1".
- [c10] The method of claim 4 including identifying data for each of said sections by said comparison of said pre- and post-expanded section documents.
- [c11] The method of claim 4 wherein said parsing said Domino document includes:
performing an iterative process to identify all sub-sections of said document, comprising:
fetching a document with all sections collapsed and identifying sections having expansion;
if level 1 section expansions are identified, fetching said document having level 1 sections expanded;
discovering and parsing new sub-sections below said level 1 sections;
if level 2 section expansions are identified, fetching said document having level 2 sections expanded;
discovering and parsing new sub-sections below said level 2 sections; and
continuing said iterative process until level N section expansions are identified and fetched.
- [c12] The method of claim 4 wherein said javascript conditionals include instructions to display section data.

- [c13] The method of claim 12 including using cookies to preserve each expansion state.
- [c14] The method of claim 13 including modifying said expansion state by having a user click on a triangular twisty.
- [c15] The method of claim 10 wherein said javascript comprises:
a session cookie remembering when said section was expanded;
a set cookie to remember when said section is collapsed;
and
a set cookie to remember when said section is currently expanded.
- [c16] The method of claim 15 further comprising a command html for a collapsed section and a command html for an expanded section.
- [c17] A method of migrating a section-rich Domino document to a non-domino web server having a client end and a server end, comprising:
performing iterative recursive intelligent fetch process steps including:
parsing said Domino document;
identifying unexpanded sections; and
constructing a URL to expand said unexpanded sections;

performing a recursive differential process step including:
inputting expanded section data at depth N expansion;
inputting expanded section data at depth N-1 expansion;
comparing pre- and post-expanded section documents through differences in html;
creating delta html files based on said comparison;
adding javascript conditionals around said delta html files; and
merging said delta html files into said depth N-1 expansion; and
expanding and collapsing said section data at said client end.

[c18] The method of claim 17 wherein said recursive intelligent process steps further comprise:
fetching a URL;
performing expansions for all depths;
parsing said document;
constructing a URL to expand unexpanded sections or fetching a new URL if no unexpanded sections exist; and
applying said constructed URL to an iterative process to expand all sections.

[c19] A program storage device readable by a machine, tangibly embodying a program of instructions executable by

the machine to perform method steps for migrating a Lotus Notes Domino document to a non-Domino web server having a client end and a server end, said method steps comprising:

performing iterative recursive intelligent fetch process steps;

performing a recursive differential process step; and
expanding and collapsing said section data at said client end.

[c20] The program storage device of claim 19 wherein said iterative recursive intelligent fetch process includes:
parsing said Domino document;
identifying unexpanded sections; and
constructing a URL to expand said unexpanded sections.

[c21] The program storage device of claim 19 further comprising:
inputting expanded section data at depth N expansion;
inputting expanded section data at depth N-1 expansion;
comparing pre- and post-expanded section documents through differences in html;
creating delta html files based on said comparison;
adding javascript conditionals around said delta html files; and
merging said delta html files into said depth N-1 expansion;

sion.

- [c22] A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for migrating a section-rich Domino document to a non-domino web server having a client end and a server end, said method steps comprising:
- performing iterative recursive intelligent fetch process steps including:
 - parsing said Domino document;
 - identifying unexpanded sections; and
 - constructing a URL to expand said unexpanded sections;
 - performing a recursive differential process step including:
 - inputting expanded section data at depth N expansion;
 - inputting expanded section data at depth N-1 expansion;
 - comparing pre- and post-expanded section documents through differences in html;
 - creating delta html files based on said comparison;
 - adding javascript conditionals around said delta html files; and
 - merging said delta html files into said depth N-1 expansion; and
 - expanding and collapsing said section data at said client

end.

- [c23] The program storage device of claim 22 wherein said recursive intelligent process steps further comprise:
- fetching a URL;
 - performing expansions for all depths;
 - parsing said document;
 - constructing a URL to expand unexpanded sections or
 - fetching a new URL if no unexpanded sections exist; and
 - applying said constructed URL to an iterative process to expand all sections.